ISSN 2229-5518

FP Tree Algorithm using Hybrid Secure Sum Protocol in Distributed Database

Jyotirmayee Rautaray, Raghvendra Kumar

Abstract— At the enhancement of new technology and growth of the network the new data is coming and being added to the database at every fraction of seconds. For accessing and storing data a specialized tool is required. An FP Tree mining algorithm increases the environmental growth for accessing the data such as customer changing environments like changing customer preference. For increasing the performance which is more reliable technique used for partitioning of the database. But if the database is partitioned that needs more security as compared to the centralized database. In this paper we mainly address the horizontally partitioned database with the help of hybrid secure sum protocol for privacy purpose.
.

Index Terms— Horizontal partitioning, Vertical partitioning, Hybrid partitioning, FP tree algorithm, Secure sum protocol, Secure multi party computation, Hybrid secure sum protocol.

1 INTRODUCTION

Data mining is wieldy used in today's scenario like genetics, mathematics and marketing etc. But the whole data is sorted according to the relationship between one data to another data and large amount of data is stored on large databases according to the database technology and data collection methods. There are mainly two type of database present in the data base scenario. One if centralized database [7] and another one is distributed database [7]. In centralized database only one server is there where all the data is stored in a central manner so that different kind of problem arises like flow control, congestion control and access control etc. But in distributed database, divide whole database into number of different server or parties so that each server have same or different database. That's why the execution of query became very fast as compared to centralized database. But the problem arises when the data is portioned because all the parties want to know the global result. So those for providing the privacy, different type of protocols are used. In this case two parties are used like secure sum protocol and for multi party used secure multi party computation. Mainly two type of models [9] [10] [11] present in distributed database scenario- one is ideal model and another one is real model. In ideal model there is no third party present. But in real model there is presence of Third party. Third party consider as a trusted third party.

1.1 Database Partitioning

Database is partitioned into three main partitioning methods e.g. horizontal partitioning, vertical partitioning and Hybrid partitioning.

1.1.1 Horizontal Partitioning

In horizontal partitioning [9] [10] [11], partition the database into row spiriting manner such that the number of attributes remain same but the number of row is different in each partition it also called homogeneous database.

1.1.2 Vertical Partitioning

In vertical partitioning [9] [10] [11], partition the database according to the column manner such that the number of transaction is same but the number of attributes is different in each partition it's also called the heterogeneous database partition.

1.1.3 Hybrid Partitioning

In hybrid partitioning [9] [10] [11], partition the databaselike first horizontal partition and then vertical partition or vice versa.

1.2 FP Tree Rule Mining Technique

The interior of mining association rule in data mining contains two main techniques. One is bread first search (BFS) and another one is depth first search (DFS). BFS uses apriori algorithm to mine the frequent item sets from a given set of data. Second one is DFS technique that is used by FP tree algorithm [1] [2] [3] [14] to mine the frequent item sets in a depth search manner. DFS contains two types of orders according to the user requirements. It may be top down or bottom up ap-

Jyotirmayee Rautaray "School of Computer Engineering, KIIT University, Odisha, India" (Jyotirmayee.1990@gmail.com)

Raghvendra Kumar "School of Computer Engineering, KIIT University, Odisha, India" (raghvendraagrawal7@gmail.com)

proach. In this paper we use top down approach of FP tree algorithm. FP tree algorithm is based on tree algorithm and this algorithm is divided into two main part- first is building FP tree and second one is mining that FP tree.

Algorithm1: - FP Tree Construction
Input- A transaction database DB with minimum support (min supp).
Method for Construction-

1. Scan the whole database DB with minimum support and set the frequent item that support minimum support and arrange that frequent item sets into a descending order and list the frequent item set.
2. Create a root node and level as a NULL.

Output- FP trees the frequent pattern of DB.

Algorithm2: -Mining Frequent Pattern with FP Tree Algorithm

Input – FP tree is constructed according to the Algorithm1 with minimum support and mine those elements that not support minimum support.
Method For Construction – Call for FP Grouth (FP Tree, NULL) which is taken from [14].
Output-complete set of frequent item sets.

1.3 Secure Sum Protocol

Secure sum protocol [5] [6] [9] is used for providing privacy to the database in case of two party .so that two parties never know the data of other parties. First party send the value to the next party after adding with own random number.

1.4 Secure Multi Party Computation

Secure multi party computation [5] [6] [9] concept came when the number of parties are greater than two and all parties wants the global result. Then all partial send their own value after adding the random number to the next party present in the database environments, and this process continues till all do not find the global result.

1.5 Hybrid Secure Sum Protocol

For providing the highest privacy to distributed database here in this paper we use hybrid based secure sum protocol [8]. Each party presents in the database in homogeneous environment and party calculate their partial support by using the following formulae (Partial Support (PS) =X. Support- Minimum support *|Size of the Database|). In hybrid secure sum protocol each party divide their data into number of segments but the number of segment are not greater than three. This is only assumption in hybrid secure sum protocol and each party selects their own user defined random number. Algorithm3 shows the step of process.

Algorithm3:- Hybrid Secure Sum Protocol

1. First each party divides the whole database into number of different data segments (Dn1……..Dn n) and each party also selects their own random number (Rn1, Rn2……R nn) according to the number of data segments.
2. Each party send the first data segment (D11, D21…….Dn1) to the third party then third party calculated the sum S after adding first data segments.
3. Then the third party sends that calculated sum to the first party.
4. Party P1 subtracts its first random number and added second random number as well as second data segments and this process continue till N=n.
5. Nth party sends that same sum to the previous party Pn-1.
6. Then Pn-1 subtract its second random number (R12, R22……..Rn2) and after that added third data segments (D13, D23.……Dn3) as well as third random number (R13, R23……..Rn3) till P1.
7. Then party P1 sends that sum to the third party.
8. Then the third party sends the sum to the Pn party.
9. Party Pn subtract its second random number and added the third data segments after that send to the previous party Pn-1.
10. Party Pn-1 subtracts its third random number (Rn3) and send to the previous party till P1.
11. Party P1 send the sum to the third party.
12. Third party broadcast to the sum to all the parties' presents in the database environments.

2. IMPLEMENTATION

This algorithm is applicable when the numbers of sites are greater than 3 and each party divides the database into number of segments and each segmented party selects their own random number. Each time parties add their next random number and next data segments but subtract their previous random number. This is applicable in homogeneous data base environments. Table 1,2,3,4 shows the different party database with minimum support 40%. Figure 1,2,3,4 shows the FP tree structure of the database.

TABLE 1
DATA SET FOR PARTY1

Transaction ID	A1	A2	A3	A4
T1	1	1	1	1
T2	1	1	0	1
T3	1	1	1	0
T4	1	1	0	1

Step1:- TID LIST OF ITMES
T1 A1, A2, A3, A4

T2	A1, A2, A4
T3	A1, A2, A3
T4	A1, A2, A4

Step2:- A1:4, A2: 4, A3:2, A4:3
Step3:- arranging in descending order
A1:4, A2:4, A4:3, A3:2
Step4:-

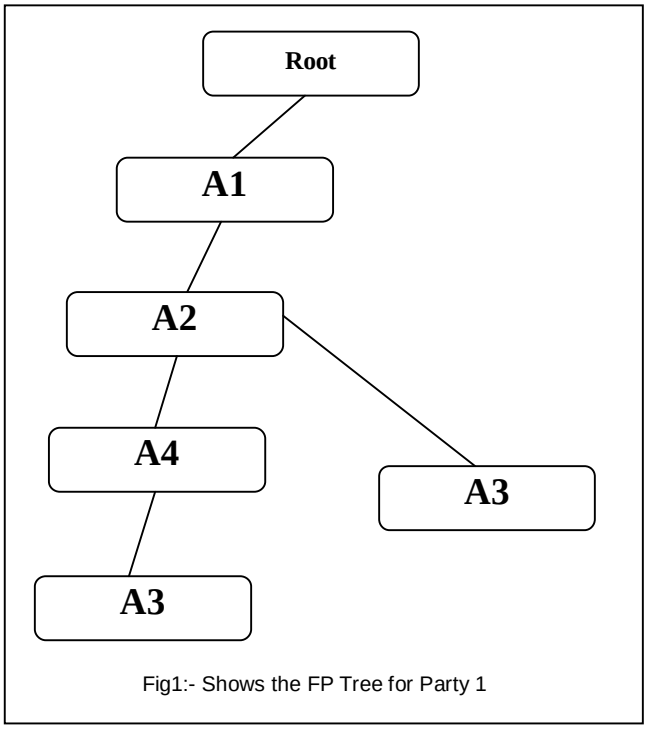

Fig1:- Shows the FP Tree for Party 1

Step5:-A1= {(A1:4)}
A2= {(A1:4)}
A3= {(A4:1, A2:1, A1:1) (A2:1, A1:1)}
A4= {(A2:3, A1:3)}
Step6:- A2= {(A2:2, A1:2)}
Step7:- A3A2:2, A3A1:2
Step8:-Support (A3A2) = Count (A3A2)/ |Total Number of Transaction| =2/4=0.5 or 50%
Support (A3A1) = Count (A3A1)/ |Total Number of Transaction| =2/4=0.5 or 50%
Candidates sets = {A1, A2, A3}

TABLE2
DATA SET FOR PARTY2

Transaction ID	A1	A2	A3	A4	A5
T1	1	1	1	0	1
T2	1	1	1	0	0
T3	0	0	1	1	1

Step1:- TID LIST OF ITMES
T1 A1, A2, A3, A5
T2 A1, A2, A3
T3 A3, A4, A5
Step2:- A1:2, A2: 2, A3:3, A4:1, A5:2
Step3:- arranging in descending order
A3:3, A1:2, A2:2, A5:2, A4:1
Step4:-

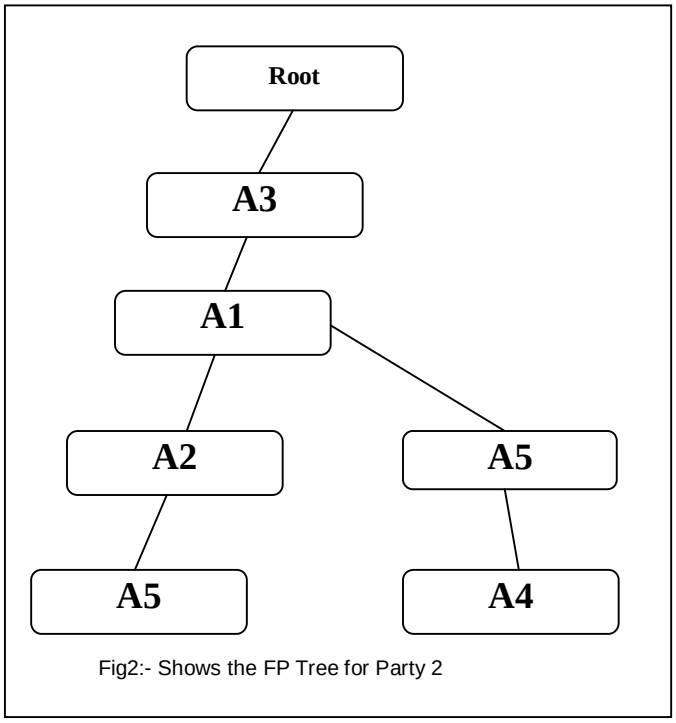

Fig2:- Shows the FP Tree for Party 2

Step5:-A1= {(A3:2)}
A2= {(A1:2, A3:2)}
A3= {(A3:3)}
A4= {(A5:1, A3:1)}
A5 = {(A2:1, A1:1, A3:1)(A3:1)}
Step6:- A5= {(A3:2)}
Step7:- A5A3:2
Step8:-Support (A5A3) = Count (A5A3)/ |Total Number of Transaction| =2/3=0.6 or 60%
Candidates sets = {A3, A5}

TABLE3
DATA SET FOR PARTY3

Transaction ID	A1	A2	A3	A4
T1	1	0	0	1
T2	0	1	1	0
T3	1	0	1	0

Step1:- TID LIST OF ITMES
T1 A1, A4
T2 A2, A3
T3 A1, A3
Step2:- A1:2, A2: 1, A3:2, A4:1
Step3:- Arranging in descending order
A1:2, A3:2, A2:1, A4:1

Step4:-

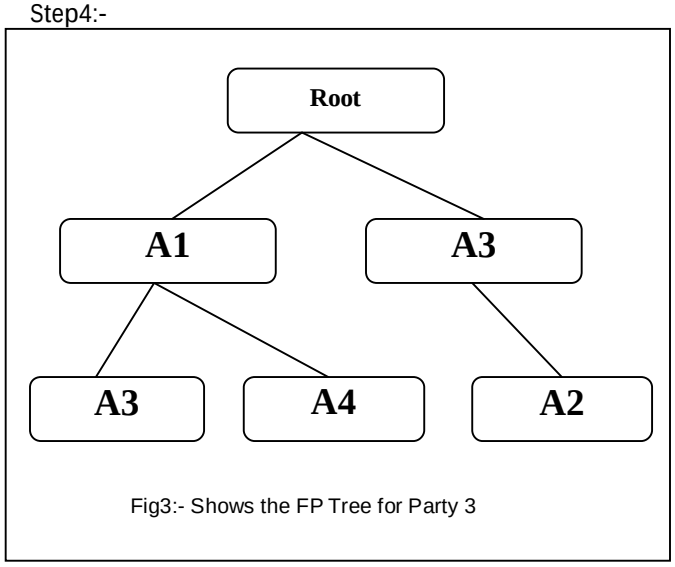

Fig3:- Shows the FP Tree for Party 3

Step5:-A1= {(A1:2)}, A2= {(A1:1)}, A3= {(A1:2) (A3:1)}
A4= {(A1:1)}
Here no candidates is selected

TABLE 4
DATA SET FOR PARTY 4

Transaction ID	A1	A2	A3	A4	A5	A6
T1	1	1	1	0	0	0
T2	1	1	1	1	1	0
T3	1	1	1	0	1	0
T4	0	1	0	1	0	1

Step1:- TID LIST OF ITMES
T1 A1, A2, A3
T2 A1, A2, A3, A4, A5
T3 A1, A2, A3, A5
T4 A2, A4, A6

Step2:- A1:3, A2: 4, A3:3, A4:2, A5:2, A6:1
Step3:- arranging in descending order
A2:4, A1:3, A3:3, A4:2, A5:2, A6:1
Step4:-

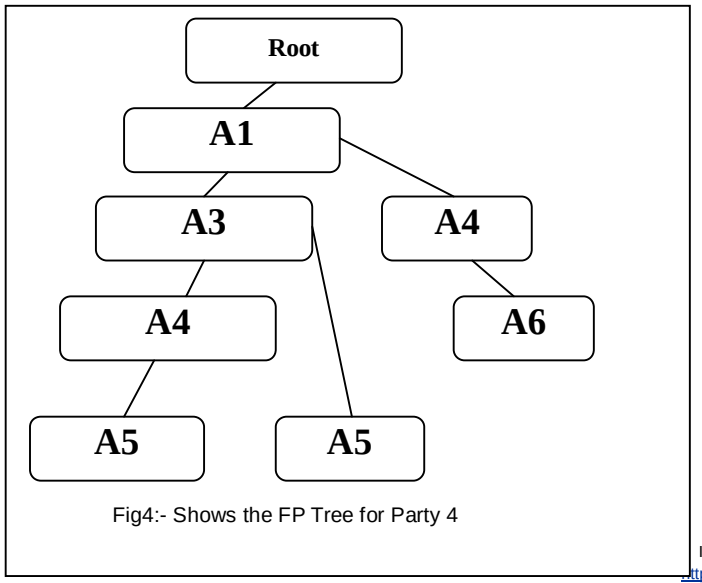

Fig4:- Shows the FP Tree for Party 4

Step5:-A1= {(A2:3)}
A2= {(A2:4)}
A3= {(A1:3, A2:3)}
A4= {(A3:1, A1:1, A2:1) (A2:1)}
A5= {(A4:1, A3:1, A1:1, A2:1) (A3:1, A1:1, A2:1)}
A6= {(A4:1, A2:1)}
Step6:- A4= {(A2:2)}
A5= {(A3:2, A1:2, A2:2)}
Step7:- A4A2:2, A5A3:2, A5A1:2, A5A2:2
Step8:-Support (A4A2) = Count (A4A2)/|Total Number of Transaction|=2/4=0.5 or 50% Support (A5A3) = Count (A5A3)/|Total Number of Transaction|=2/4=0.5 or 50%
Support (A5A1) = Count (A5A1)/|Total Number of Transaction|=2/4=0.5 or 50%
Support (A5A2) = Count (A5A2)/|Total Number of Transaction|=2/4=0.5 or 50%
Candidates sets = {A1, A2, A3, A4, A5}
At party 1 Candidate's item sets {A1, A2, and A 3}
At party 2 Candidate's item sets {A3, A5}
At party 3 has no Candidate's item sets is selected
At party 4 Candidate's item sets {A1, A2, A3, A4, A5}
Let us consider the item set = {A1}
Each party calculated their partial support by using following formula
Partial Support (PS) = X. Support- Minimum Support*|Size of the Database|
At party 1 partial support (PS1) =4-0.4*4=2.4
At party 2 partial support (PS2) =2-0.4*3=0.8
At party 3 partial support (PS3) =2-0.4*3=0.8
At party 4 partial support (PS4) =3-0.4*4=1.4
Party1 divide the partial support (Data) into number of different segments
PS11=1.0, PS12=1.0, PS13=0.4
Party2 divide the partial support (Data) into number of different segments
PS21=0.4, PS22=0.4, PS23=0.0
Party3 divide the partial support (Data) into number of different segments
PS31=0.0, PS32=0.4, PS33=0.4
Party4 divide the partial support (Data) into number of different segments
PS41=0.4, PS42=1.0 PS=0.0
Let's consider a Random number (user defined number) for each data segments
RN11=1, RN12=1, RN13=0.0
RN21=2, RN22=1, RN23=1
RN31=1, RN32=0.0, RN33=2.0
RN41=1, RN42=2
At first time each party send their first data segment (D11, D12…….d1n) and first random (R11, R21……..Rn1) to the Third Party third party third party calculated the sum after adding their first data segments
Sum1=D11+D12+……D1n
Sum1=1.0+0.4+0.0+0.4=1.8
After that Third Party sanded the result to party P1 S=1.8

After that party P1 take that sum and subtract its first random number and added its second random number as second data segment after that sanded to the next party till Pn
Sum (Sum1) = Sum1-Rn1+Rn2+Dn2
Sum (Sum1)1 = 1.8-1+1+1=2.8
Sum (Sum1)2=2.8-2+1+0.4=2.2
Sum (Sum1)3=2.2-1+0.0+0.4=1.6
Sum (Sum1)4=1.6-1+2+1.0=3.6
After that Pn send that sum to the Pn-1 party then Pn-1 subtract its second random number and added its third random number as well as third data segments till P1
Sum (Sum1 (Sum2)) 3= Sum (Sum1) - Rn2+Rn3+Dn3
Sum (Sum1 (Sum2)) 33= 3.6-0.0+2.0+0.4=6.0
Sum (Sum1 (Sum2)) 32=6.0-1.0+1.0+0.0=6.0
Sum (Sum1 (Sum2)) 31=6.0-1.0+0.0+0.4=5.6
After that P1 send that sum to the third party and third party replied that sum to the Pn Party then Pn party added its third data segments and send to the previous party till P1 but the Pn-1 party only subtract their Rn3 to the sum then result calculated by Pn using the following formula
Sum ((Sum1 (Sum2)) sum3)44 = Sum (Sum1 (Sum2)) 3+Dn3
Sum ((Sum1 (Sum2)) sum3)44=5.6+0.0=5.6
Pn-1 calculated the by using the following formula
Sum ((Sum1 (Sum2)) sum3)43= Sum ((Sum1 (Sum2)) sum3)44-Rn3
Sum ((Sum1 (Sum2)) sum3)43=5.6-2.0=3.6
Sum ((Sum1 (Sum2)) sum3)42=3.6-1.0=2.6
Sum ((Sum1 (Sum2)) sum3)41=2.6-0.0=2.6
After that P1 sanded the Sum ((Sum1 (Sum2)) sum3)41 to the third party and then third party will broadcast that result to all the party presents in the homogeneous database environment. All parties declare the global excess support is 2.6.

3 CONCLUSIONS

In this paper we suggest new approach for privacy preservation in FP tree algorithm. This is combination of FP tree algorithm and hybrid secure sum protocol. For providing the highest privacy to the homogenous database environment. This is applicable for the multi party environments when party is greater than two. And in future implement the highest privacy preserving rule mining technique for FP tree algorithm.

REFERENCES

[1].Agrawal, R., et al.: Mining association rules between sets of items in large database. In: Proc. of ACM SIGMOD'93, D.C, 1993,pp.207-216 ACM Press, Washington.

[2]. Agarwal, R., Imielinski, T., Swamy, A.: Mining Association Rules between Sets of Items in Large Databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 1993, pp. 207-210.

[3]. Srikant, R., Agrawal, R.: Mining generalized association rules. In: VLDB'95,1994, pp.479-488 .

[4]. Agrawal, R., Srikant, R.: Privacy-Preserving Data Mining. In: proceedings of the 2000 ACM SIGMOD on management of data, 2000, pp. 439-450.

[5]. Lindell, Y., Pinkas, B.: Privacy preserving Data Mining. In: Proceedings of 20th Annual International Cryptology Conference (CRYPTO) 2000.

[6]. Kantarcioglu, M., Clifton, C.: Privacy-Preserving distributed mining of association rules on horizontally partitioned data. In IEEE Transactions on Knowledge and Data Engineering Journal, IEEE Press, Vol 16(9), 2004, pp.1026-1037.

[7]. Han, J. Kamber, M.:Data Mining Concepts and Techniques. Morgan Kaufmann, San Francisco, 2006.

[8]. B., Mishra, "Hybrid technique for secure sum protocol". WCSIT, Vol 1, 2011, pp.198-201.

[9]. Sugumar, Jayakumar, R., Rengarajan, C.:Design a Secure Multi Site Computation System for Privacy Preserving Data Mining. In International Journal of Computer Science and Telecommunications, Vol 3, 2012, pp.101-105.

[10]. Muthu Lakshmi, N. V., Sandhya Rani, K.: Privacy Preserving Association Rule Mining without Trusted Site for Horizontal Partitioned database. In International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.2, 2012, pp.17-29.

[11]. Muthu lakshmi, N.V., Sandhya Rani, K.: Privacy Preserving Association Rule Mining in Horizontally Partitioned Databases Using Cryptography Techniques. In International Journal of Computer Science and Information Technologies(IJCSIT), Vol. 3 (1),2012 , PP. 3176 – 3182.

[12]. Goldreich, O., Micali, S. & Wigerson, A.: How to play any mental game. In: Proceedings of the 19th Annual ACM Symposium on Theory of Computing, pp.218-229.

[13]. Franklin, M., Galil, Z. & Yung, M.:An overview of Secured Distributed Computing. Technical Report CUCS- 00892, Department of Computer Science, Columbia University.

[14]. Dehao C. "Tree Partition based Parallel Frequent Pattern mining on Shared Memory Systems" IEEE.